BELLCOMM, INC.

COVER SHEET FOR TECHNICAL MEMORANDUM

TITLE- Status Report on Countdown Simulation

TM- 64-1031-3

FILING CASE NO(S)- 140

DATE- December 17, 1964

AUTHOR(S)- E. B. Parker, III
P. S. Schaenman

FILING SUBJECT(S)- (ASSIGNED BY AUTHOR(S)) Countdown Simulation
Simulation

FACILITY FORM 602

[redacted] X67-89311
(ACCESSION NUMBER) (THRU)
21 0
(PAGES) (CODE)
[redacted] 11
(NASA CR OR TMX OR AD NUMBER) (CATEGORY)

ABSTRACT

Many of the problems associated with countdown planning require a sensitivity analysis of the probability of meeting a launch window as a function of various factors. These factors include:

a. logical sequence of activities
b. equipment failure rate distributions
c. equipment repair time distributions
d. recycle policies
e. length of launch window

A tool for performing such an analysis has been developed at Bellcomm in the form of a countdown simulation programmed for use on a digital computer. The simulation takes the above factors as inputs and performs a large number of Monte Carlo trials of the countdown.

The prime output of the simulation is the distribution of times to complete the countdown. The probability of meeting a launch window is then the fraction of trials which were successfully completed within that launch window.

The Bellcomm effort has thus far demonstrated the feasibility of simulating a countdown from a systems viewpoint. It has also shown that the simulation can be made fast enough to yield statistically meaningful results within reasonable amounts of machine time. The most immediate problem facing potential users is the difficulty in collecting reasonable estimates of the inputs, particularly a PERT-like diagram of the countdown.

JAN 1978

[redacted] N79-72059

(NASA-CR-155644) STATUS REPORT ON COUNTDOWN SIMULATION (Bellcomm, Inc.) 22 p

Unclas
00/14 12578

BELLCOMM, INC.

TM- 64-1031-3

DISTRIBUTION

COMPLETE MEMORANDUM TO

CORRESPONDENCE FILES:

OFFICIAL FILE COPY
plus one white copy for each additional case referenced

TECHNICAL LIBRARY (4)

MSF

J. K. Holcomb/MAO
T. A. Keegan/MA-2
J. F. Underwood/MAT

KSC

R. E. Moser/VT
G. F. Williams/VE

MSFC

C. G. Jones/R-PVE-VOR
L. G. Richard/R-ASTR-S

MSC

R. W. Lanzkron/PS
W. A. Lee/PA
K. L. Thomas/EA-2

Bellcomm

G. M. Anderson
C. Bidgood
W. E. Boyes
T. J. Celi
G. W. Craft
J. P. Downs
S. G. Embrey
J. S. Engel
E. L. Gruman

COVER SHEET ONLY TO

Bellcomm (Continued)

J. A. Hornbeck
J. Jacobs
L. Jarrell
B. R. Kahan
W. M. Keese
C. M. Klingman
P. R. Knaff
J. Kranton
B. H. Liebowitz
A. L. Locicero
C. A. Lovell
W. J. Martin
L. G. Miller
V. Muller
I. D. Nehama
J. M. Nervik
M. A. Robinson
I. M. Ross
A. H. Scheinman
S. J. Schoen
T. H. Thompson
J. M. West
Department 2014

CASE FILE COPY

I. General Description of Simulation

The simulation consists of making a large number of Monte Carlo countdown trials using a mathematical-logical model of the countdown. The probability of meeting a launch window is the fraction of trials in which the countdown is successfully completed within that launch window.

A program to run this simulation on a digital computer has been written in MAP and FORTRAN IV. It may be run on an IBM 7040 or 7044 computer operating under an IBSYS Monitor System, or on the Bellcomm 7040-44 computer system using BCMSYS.

Inputs to the simulation - the factors which determine the launch probability - include:

a. the logical sequence of countdown activities (i.e., a PERT-like diagram of the countdown)

b. a list of the equipments needed to perform each activity ("equipments" include S/C systems, L/V systems, range systems and GSE systems)

c. the failure rate and repair time distributions for each equipment

d. the decision policy to be used for handling failures and countdown termination

e. the length of the launch window

For each countdown trial the simulation operates by looking at the countdown whenever a "significant event" occurs. "Significant events" are the start of an activity, the successful completion of an activity, the arrival of an equipment failure, and the completion of repairs on an equipment.

Each type of significant event has its own method of being processed. However, all of the processing methods consist of updating equipment histories, changing activity and equipment states, and predicting when other significant events will take place. The countdown trial proceeds until either the countdown is successfully completed or a decision to scrub is made.

[redacted]

When a countdown trial ends, the desired data (e.g., completion time, equipment usage times, scrub decisions) are recorded and collated with data from previous trials. The states and histories of equipments and activities are reset and a new trial begun if requested.

Among the problems associated with using digital computer simulations are computer usage time and computer memory requirements. Both were basic considerations in setting up this simulation and have turned out to be within workable limits. The simulation can handle hundreds of activities and hundreds of equipments with running times* per countdown trial (after setting up the simulation on the machine) varying from 0.5 - 5.0 sec. The variations in time are caused by the number and values of the inputs, and by the number and form of the outputs requested.

Another problem associated with this particular simulation, is the difficulty in obtaining the necessary inputs. This problem is not one of the inputs not existing, but rather one of the inputs not being on paper. For instance, the authors have been unable to find PERT-like diagrams of countdowns. The information in a countdown manual is insufficient - it does not show logical relationships, i.e., which activities must be done in series and which may be done in parallel. Bounds on the other inputs can be estimated, but parametric studies cannot be run without these PERT-like diagrams. Producing a PERT-like diagram of a Saturn countdown would involve having the personnel associated with planning the countdown document the countdown information in this form.

It was intended that the present version of the simulation:

a. prove the feasibility of making a systems analysis of the countdown using Monte Carlo techniques

b. determine if such techniques could be used in reasonable amounts of computer time, and

c. serve as a basic model which can be further sophisticated without complete revision.

These goals have largely been satisfied.

---

*These running times refer to an IBM 7040-44 computer system operating under BCMSYS.

II. Inputs

The basic inputs to the simulation are the logical sequence of activities which make up the countdown, the equipments needed to perform these activities, the length of the launch window, and the decision policy for handling failures and countdown termination. This section defines these inputs and gives their computer memory space requirements.

A. Activities

Activities are defined here as they are in PERT: those mutually-exclusive tasks which require manpower, material, facilities, or other resources and which together constitute the countdown. Each activity A_{ij} is characterized by the seven items described below. Items 1 - 3 must be specified if they exist. Items 4 - 7 must exist and must be specified.

1. Immediate Logical Prerequisities (ILP). The immediate logical prerequisites are those activities which must be successfully completed immediately before A_{ij}. "Immediately" refers to logic rather than time. Example: Referring to Figure 1, where arrows represent activities and circles the beginnings of activities, the only ILP's of A_{45} are A_{24} and A_{34}. A_{12} and A_{13}, although prerequisities, are not immediate prerequisites.

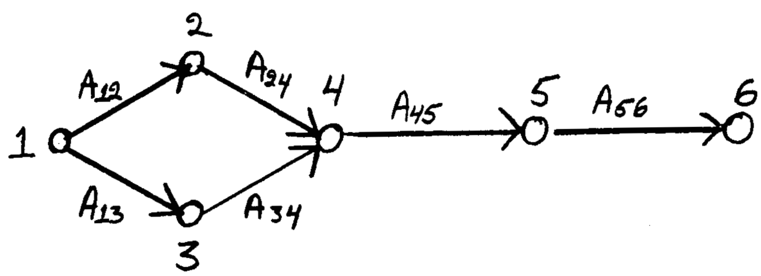


Figure 1 - Activity Sample Sequence

2. Immediate Logical Followers (ILF). The immediate logical followers are those activities for which A_{ij} is an immediate logical prerequisite. In Figure 1 the ILF for A_{45} is A_{56}.

Note that specifying items 1 and 2 for every activity is equivalent to giving the PERT network of the countdown.

3. Time Prerequisite. A time prerequisite is the "real" time before which activity A_{ij} cannot start. "Real" time is counted continuously from zero starting at or before the first activity, as opposed to T-time which is counted backward to zero at lift-off and which is suspended during holds.

   Time prerequisites can be used to simulate built-in holds, as in the example below. With minor program modification they may also be used to simulate prerequisites in T-time. Example: Referring to Figure 2, assume that activity A_{12} starts at real time t=0 and that it normally takes 10 minutes to be completed. A time prerequisite of t=40 on activity A_{23} would then represent a 30 minute built-in hold between the activities.

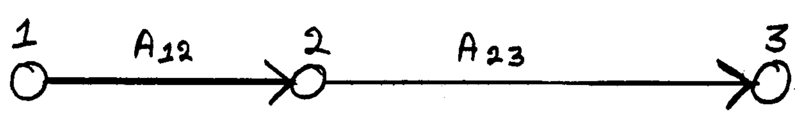


Figure 2 - Sample Activity Sequence

4. Equipments. Equipments are those physical items needed to perform activity A_{ij} (see more complete definition in section II-B).

5. Activity time. The activity time is the length of time needed to perform activity A_{ij} if none of its equipments fail. It is specified as either a constant or a random variable drawn from a given distribution. Note that "activity time" is not, in this version, permitted to be a function of the length of the countdown.

6. Mode Time. Mode time is the time that each equipment required by A_{ij} is in a particular mode during A_{ij}. The mode times are given as constants which sum to the mean activity time. When random activity times are used these constant mode times are adjusted to sum to the random number chosen as the activity time.

7. Recycle Points. When activity A_{ij} is interrupted by the failure of one of its equipments, the activity is stopped while the equipment is restored to working condition. When restoration is completed, certain activities are recycled to, i.e., become the ILF's of the restoration process. These activities are then called the "recycle points" of A_{ij}. At least one recycle point must be specified for each activity. More will be said about them in section II-D.

B. Equipments

Equipments are those mutually exclusive physical items needed to perform the various activities of the countdown. Equipments may be parts of the space vehicle, GSE, or range. They may be components, systems or subsystems. The only constraints are that each must be associated with at least one activity and must be further defined by the following information, all of which must be given for each equipment:

1. Modes of Operation. The various levels of usage of the equipment; e.g., off, warm-up, full-on.

2. Time-to-first-failure Distribution Function Per Mode (TTFF-d.f.). The TTFF-d.f. is the relationship between time t_i and the probability that the equipment fails by time t_i, where t_i is time accumulated in mode i and is measured from the latest renewal point of the equipment. Renewal

points are either the start of the countdown or the completion of repairs on the equipment. A time-to-first-failure distribution function must be given for each mode.

3. Restoration Time Distribution Function. Restoration time is the time needed to restore an equipment to working condition. It is measured from the time the equipment was discovered to have failed and includes time needed for diagnosis, access, and repair or replacement. It does not include time needed to repeat a portion of the countdown. The restoration time is a random variable; its distribution function is given as an input.

C. Launch Window and Upper Bound

A launch window is the continuous interval of time within which the countdown must be completed if a launching is to be made. The length of such a window is supplied as an input. Since some countdown trials are allowed to run beyond the end of the launch window to see when they would have finished, an upper bound on the length of a trial must also be given as an input.

D. Decision Policy for Handling Failures and Countdown Termination

In this version of the simulation the authors have chosen to implement only one of the many possible policies for handling failures and deciding when to terminate the countdown. The only degree of freedom left to the user is the choice of recycle points. Later versions of the simulation will hopefully provide additional degrees of freedom in this policy and make it more justifiable to call the policy an "input". The present policy is as follows:

1. Equipment Failures

   a. Failures are detected immediately if the equipment is in use, or immediately upon attempting to use the equipment if it is not in use at the time of failure.

   b. As soon as failure is detected, the activity involved is stopped and the faulty equipment immediately enters a restoration activity. Those activities for which the interrupted activity is a prerequisite are held up, but no others are delayed.

c. When the equipment has been restored to working order, the interrupted activity's recycle point list is used to determine where to recycle to; i.e., which activities are to be the immediate logical followers of the restoration process.

2. Terminating the Countdown

Following the completion of every activity and restoration process, a check is made to see if the length of the critical path[1] to the end of the countdown exceeds the remaining time to the end of the launch window. If it does not, the countdown trial continues (unless, of course, this was the last activity). If it does, a decision to terminate the countdown is recorded but the simulated countdown continues until it is either successfully completed or exceeds the upper bound.

E. Memory Space Requirements

The memory size of the digital computer which is used to run this simulation limits the numbers of events, equipments, and other inputs which can be specified[2]. The program itself uses about 8K words. In addition, the number of spaces needed per equipment is

$$[4 + 6N]$$

and the number per activity is

$$[2(R + P + F) + L + Q\ (N + 2) + 14]$$

where R = no. of recycle points

P = no. of ILP's

F = no. of ILF's

L = no. of launch windows being considered

Q = no. of equipments

N = no. of modes

---

[1]The "critical path" is defined here as the path (through the logical sequence of remaining activities) with the longest mean time. The mean time for each path is calculated assuming no further failures will occur.

[2]Tapes may be used to augment the memory of the computer but the increase in running time is felt to be prohibitive for purposes of this simulation.

If the average value of each of these parameters were 3, the Bellcomm 7040-44 computer system (with 32K memory) could easily handle 250 activities and 500 equipments.

III. Outputs

The particular information which is recorded and collated depends upon the user's application. Examples of available outputs are the distribution of times to complete the countdown, the equipment usage times, equipment failures and times of these failures, and a record of decisions to prematurely terminate the countdown.

IV. Operation of the Model

This section presents a description of how the countdown model operates. Emphasis has been placed on the overall logic rather than on the programming details. The section ends with a discussion of the machine time needed per countdown trial.

A. Definition of States and Events

Some further definitions are necessary before proceeding with the operation of the model.

1. States of an Activity

Each activity is required to be in one of the following mutually exclusive and exhaustive states at any point in time:

a. Logically Not Ready

At least one of its immediate logical prerequisites is not completed.

b. Logically Ready

All logical prerequisites are completed. The activity is scheduled to start at the time the last logical prerequisite was completed, or at its time prerequisite, whichever is latest. This "start" time is stored with the state of the activity. If at the scheduled starting time any equipment required by the activity is not available, the start time is changed to the earliest time that all equipments will be available.

c. In Progress - Success Predicted

The activity is in progress and a successful completion is predicted for it at some particular time. This "end" time is stored with the activity state.

d. In progress - Failure Predicted

The activity is in progress and at least one of its equipments is predicted to fail at some particular time. This "fail" time is stored with the activity state.

e. Limbo

The activity has been stopped because of an equipment failure. Recycling is predicted to take place at the time the faulty equipment will have been restored to working order. The "recycle" time is stored with the activity state.

f. Failed and Recycled

The activity was stopped because of an equipment failure which has since been corrected. Recycling has taken place but the activity has not been attempted again.

g. Successfully Completed

The activity has been successfully completed.

At the start of the countdown trial every activity is in either state a or b.

2. States of an Equipment

Every equipment is required to be in one of the following four mutually exclusive and exhaustive states at any point in time:

a. Not In Use

The equipment is in working condition but not being used in any activity at the moment.

b. Not In Use - Failed

The equipment has failed while not being used by any activity. The failure will not be discovered until some activity attempts to use the equipment.

c. In Use

The equipment is working and being used by some activity. [An equipment cannot be used by more than one activity at any point in time.]

d. Restoration

The equipment has had a failure which was discovered. Restoration to working order is now in progress.

At the start of the countdown trial every equipment is in the "not in use" state.

3. Events

An event is defined as the change of state of an activity or equipment. Of the many possible types of events four are considered "significant", since they trigger all other events in this simulation. [The triggered events are assumed to occur simultaneously with the significant event which triggers them].

The four "significant" events are:

a. START--the start of an activity

b. END--the successful completion of an activity

c. FAIL--the discovery of an equipment failure

d. RECYCLE--the completion of an equipment restoration and the immediate use of the recycle policy.

Note that four of the seven activity states, namely b,c, d, and e, forecast the time at which one of the above significant event will take place.

B. Preparations for a Run

A run here means a set of simulated countdown trials using a fixed set of inputs. For any run the computer first organizes the input data into various lists. If the data is invalid or ambiguous, diagnostic messages are printed and the run is terminated.

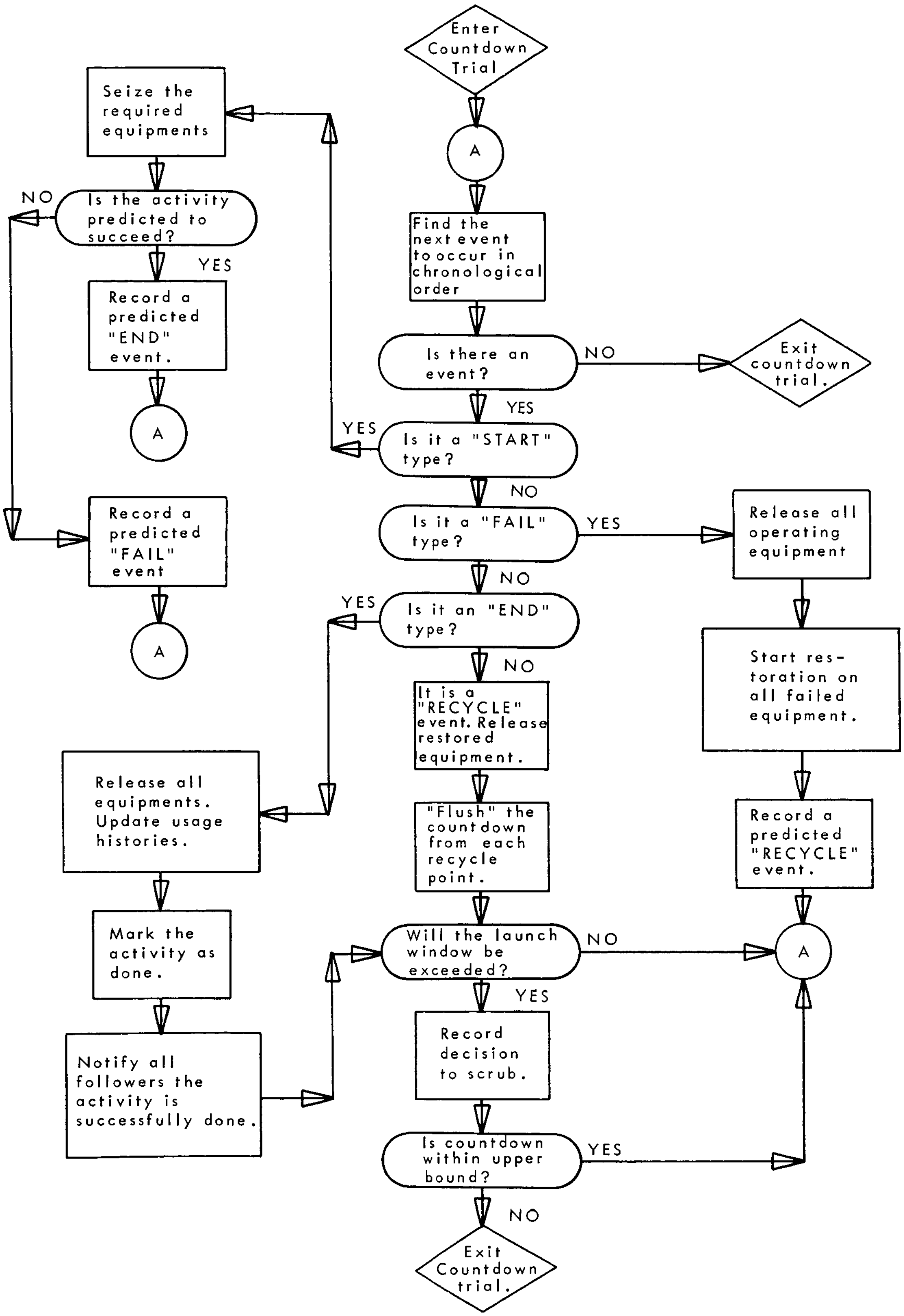
After the lists are formed, the latest allowable completion (LAC) time is computed for each activity A_{ij}. The LAC time is computed by first finding the critical path from among all paths which start with A_{ij}. Then the length of this path is subtracted from the time of the end of the launch window to get the LAC time. (If the activity is not completed by this time, a decision to terminate the countdown is recorded.) The above computations are repeated using the upper bound in place of the end of the launch window. (This LAC time becomes applicable when the countdown is allowed to run past the end of the launch window.)

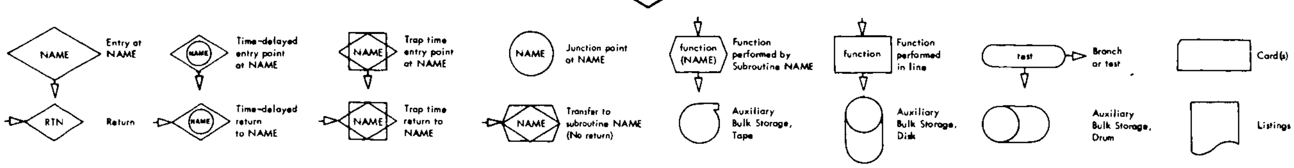
Next, the equipment usage times for a perfect (no failures) countdown are calculated and a test is made to make sure that the length of this "perfect" countdown is less than the upper bound. The simulation is then ready to start its first countdown trial. The times-to-first-failure for every mode of every equipment are generated using the distributions given in the inputs, and the simulated countdown trial begins.

C. The Countdown Trial

The countdown trial is illustrated in flow chart form in Figure 3. Each trial consists of looking at the countdown whenever a significant event occurs, and then updating the states and histories of the various activities and equipments affected by that event. The events are examined in chronological order. The "next" event to be processed is found by scanning the activity list for the earliest forecast event. The event chosen is then processed according to its type, using one of four procedures (described below). After the processing of the event is completed, the simulation returns to scanning the activity list for the next event. This procedure is repeated until either no more events are scheduled - in which case the countdown has been successfully completed - or the critical path to the end of the countdown is greater than the time to the upper bound - in which case the countdown trial is scrubbed. In either case the package of information collected during the trial is collated with data from previous trials. All states and histories are reset and a new trial begins if requested.

BELLCOMM, INC., 1100 17th ST., WASH. 36, D. C. 12/17/64





The procedures for processing each of the four types of significant events are explained in the next four subsections. The activity whose state forecasts the event is referred to below as activity A_{ij}, and the time at which the event occurs is called time t.

1. START Event

The selection of a START event as the "next" event implies that at time t activity A_{ij} has all its pre-requisites satisfied and all its necessary equipments free. These equipments are then seized - put in an "in-use" state - as of time t and a test is made to determine if any have failed since last use. Another test is made to determine if any will fail during activity A_{ij} (assuming the activity is not interrupted by external factors). Both of these tests are made using the times-to-first-failure generated at the last renewal point of each equipment. If either test turns up a failure, the state of activity A_{ij} changes from "logically ready" to "in progress-failure predicted". A fail event is scheduled for t_f, the time the failure is discovered. For failures which occurred before the start of activity A_{ij}, $t_f = t$. For failures which are predicted to occur during A_{ij}, t_f is the time of the earliest failure.

If no failure is predicted, the activity changes state from "logically ready" to "in-progress-success predicted", and an END event is scheduled for time τ, where $\tau = t +$ activity time.

After the above operations are completed, control returns to the event selection mechanism which hunts for the next event.

2. END Event

The selection of an END event implies that at time t activity A_{ij} is successfully completed. The equipments used by A_{ij} are released as of time t and their usage histories updated. The activity's state is changed from "in progress-success predicted" to "successfully completed". Each immediate logical follower of A_{ij} is notified that one of its prerequisites has been completed. If A_{ij} is the last logical prerequisite which remained to be satisfied, the follower's state is changed to "logically ready".

A check is then made to determine if A_{ij}'s LAC time corresponding to the given launch window has been exceeded. If it has been exceeded, a decision to scrub is recorded, but the trial continues unless the LAC time corresponding to the upper bound has also been exceeded. After the above operations are completed, control returns to the event selection mechanism.

3. FAIL Event

The selection of a FAIL event implies that at time t activity A_{ij} is interrupted by the failure of at least one of its equipments. The activity is stopped and its state changed from "in-progress-failure predicted" to "limbo". The failed equipments immediately enter a restoration process, i.e., changes state from "in-use" to "restoration". A recycle event is scheduled for the time that restoration of the equipment will be completed. The equipment usage times are updated and new times-to-first-failure are generated for it. The non-failed equipments are released - change state from "in-use" to "not in-use" - and their usage times are updated. Control is then returned to the event selection mechanism.

4. RECYCLE Event

The selection of a RECYCLE event implies that at time t restoration is completed on the equipment which failed during A_{ij}. The equipment's state is therefore changed from "restoration" to "not-in-use", and the activity's state from "limbo" to "failed and recycled". Recycling is immediately begun by looking up the recycle points of activity A_{ij} and setting these activities and all activities dependent on them as undone. (This undoing process is called "flushing" the countdown from the recycle points.) Any of these activities which were in progress are stopped and their equipments are released after updating histories. The recycle points themselves become either "logically ready" or "logically not ready", depending on whether or not their prerequisites are satisfied. All activities which have one of these recycle points as a prerequisite become "logically not ready". No other activities are affected.

While performing the above operations, a check is made on each recycle point to determine if its latest allowable completion time corresponding to the given launch window has been exceeded. If it has, a decision to scrub is recorded, but the trial continues unless the LAC time corresponding to the upper bound is also exceeded. After the above operations are completed, control returns to the event selection mechanism.

As an example of recycling and flushing, consider the activity sequence in Figure 4.

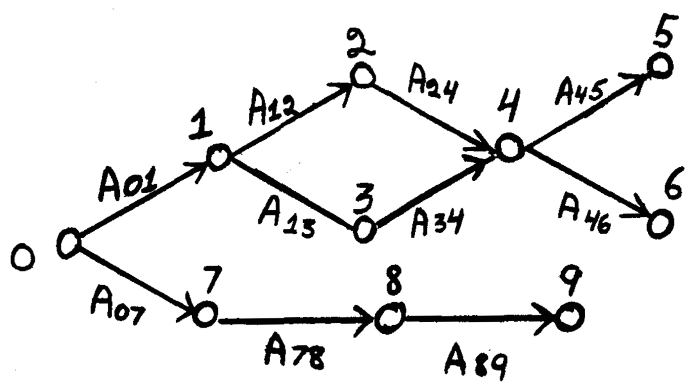


Figure 4 - Sample Activity Sequence

Suppose that an equipment failed during activity A_{45} and that A_{45}'s recycle points are activities A_{12} and A_{13}. Then recycling will cause activities A_{12}, A_{13},

A_{24}, A_{34}, A_{45}, and A_{46} all to be set undone. Activities A_{78} and A_{89} will proceed unaffected unless the countdown is scrubbed by the recycle policy.

D. Machine Time

A major requirement on setting up the simulation was to obtain fast enough countdown trials so as to permit a sufficient number of trials for statistically meaningful results within reasonable amounts of machine time. The time per countdown trial is highly dependent on the number of inputs, the values of the inputs, and on the amount of output information which is collected and printed. For example, the order of magnitude of the time per trial for a countdown involving 125 activities and 250 equipments varied from $\sim$0.75 sec. to $\sim$2.0 sec. as the MTBF of each equipment was decreased by one order of magnitude. [This resulted from a greatly increased number of failures.] The time per trial doubled when numerous outputs were printed out individually for each trial. These times are in addition to a set-up time of 12.5 minutes for the above runs. It is expected that this set-up time can be considerably reduced.

No attempt has been made yet to streamline the simulation for speed. However, three features have been devised to aid in reducing the time for a run. The first is a test before each trial to determine if it will have any failures. If it will not, the trial is immediately recorded as a success and the next trial begun. If it will have at least one failure, normal processing is used.

The second time-reducing feature is a continual watch on the mean and variance of the countdown completion times. When the rate of change of these parameters falls below pre-set bounds, the run is terminated. This test is intended to dynamically determine how many runs are needed for a particular set of inputs. A fixed upper bound on the number of trials is included to prevent any runaway situations.

The third is the ability to play the countdown against several launch windows of different lengths simultaneously. The effects of launch window length on termination decisions can thus be studied without having to make a new run for each window while holding other inputs constant.

V. Constraints and Improvements

In the course of setting up the simulation constraints were placed on various inputs for the sake of simplicity, memory space, or machine running time. Many of the constraints can be worked around by proper manipulation of the existing model (though often inefficiently) or by programming changes. This section discusses several constraints on the present version of the simulation which might be relaxed in later versions. All the suggested improvements are considered feasible within the present framework, and some have been partially implemented already.

A. Immediate Logical Prerequisites

In the current simulation the immediate logical prerequisites of an activity are AND'ed logically; all must be satisfied before the activity can start. Corequisites, i.e., activities which must start simultaneously, can also be handled using this AND logic. OR prerequisite logic, in which the activity can start if any one of several groups of ILP's are satisfied, is not available but might be useful for such things as recycling to non-normal activities.

For example, referring to Figure 5, suppose one wanted to replace activity A_{23} by activity A'_{23}

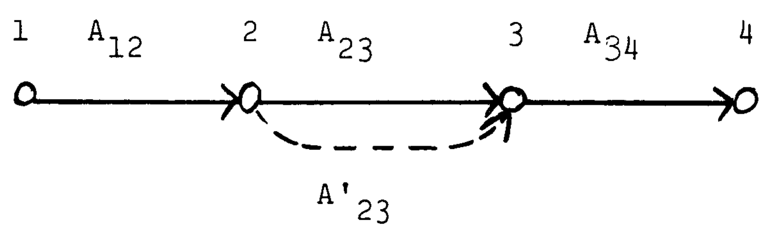


Figure 5 - Sample Activity Sequence

if A_{23} fails. (This is not possible at present.) Then A_{34} must have as its ILP A_{23} OR A'_{23}.

B. Recycle Points

1. Recycle points are restricted to activities which are part of the normal countdown, i.e., those which would be done even in a no-failure countdown. There are times, however, when recycling to "non-normal" activities such as unloading propellants is necessary before returning to the normal sequencing.

2. The recycle points of an activity are also not allowed to have that activity as a prerequisite. This eliminates the possibility of recycling forward when one wants to by-pass non-essential activities in order to save them.

3. Recycling is currently made to the beginning of the recycle point activities. But there are numerous activities such as loading propellants, which if at all possible are continued from the point of interruption.

It is obviously desirable to remove all three of the foregoing constraints.

C. Restoration

The time needed to restore an equipment in the space vehicle may be dependent on whether the arming tower has been removed, whether propellant loading has taken place, and whether the astronauts are on-board. Different restoration time distribution functions may therefore be needed for different T-time intervals. This feature is easy to add to the simulation; the only important cost is the additional memory space needed to store the extra distributions.

Several assumptions concerning restoration and not previously mentioned are the following: no failed equipment can fail again while undergoing restoration; the restoration time is not added to the equipment usage time; and the restoration process does not induce failures in other equipments.

D. Failure Detection

The simulation tries to determine the probability of meeting a launch window but does not explicitly concern itself with whether this is done with any undiscovered failures. The probabilities of detecting failures, given you are looking for them, were considered too difficult to obtain to be worth including at the present time.

E. Dealing with Detected Failures

The present method of dealing with detected failures can be improved by providing a richer choice of strategies and by increasing the number of points at which decisions are made. When a failure is discovered the range of decisions can be expanded to include whether or not to scrub, whether to hold the entire countdown or just the affected activities, whether to use the normal sequence of activities or some special sequence such as skipping the interrupted activity altogether, and whether or not to repair the faulty equipment. Any of these decisions may be treated as a function of activity and equipment states, size of and distance from the launch windows, or any other parameters of the simulation. If the countdown is not scrubbed, the same set of questions may be repeated as the holds or repairs progress, and again when they are completed. The choice of strategies is also widened if the constraints on recycle points (discussed in V-B) are lifted.

F. Termination Policy

The termination policy might be generalized so that a decision to scrub is made whenever the probability of meeting the launch window falls below a pre-set level. The latter policy implies the ability to calculate that probability as each countdown trial is running.

G. Human Factors and Weather

Human factors and weather are not explicitly considered in the simulation, but both may be treated as equipments with failure rates. Human factors can be handled by estimating the number of man-hours involved in each activity and considering that time as being spent in the "on" mode for humans. Weather can be an equipment that is "on" and continuously monitored with a failure rate reflecting both the weather constraints on the launch and the prevailing weather at the KSC. Note, however, that the present version does not allow activity time to be a function of the length of the countdown. Consequently, weather would not be monitored beyond the length of the nominal countdown.

VI. Concluding Remarks

As stated in the introduction, this simulation has thus far demonstrated that a countdown can be simulated from a systems viewpoint and that the simulation can be fast enough to economically yield statistically meaningful results. The present problem concerning its use is not simulation techniques, but rather the gathering of the necessary input data.

Future efforts will be concentrated in the following areas:

1. Obtaining the inputs necessary for using the simulation to aid in countdown planning.

2. Further developing analytical techniques to supplement the present simulation.

3. Participating in discussions with the NASA Centers in an attempt to coordinate the efforts presently being expended on developing and improving countdown models and simulations.

VII. Acknowledgment

The authors take this opportunity to express their appreciation for the programming, documentation, and editorial assistance so cheerfully provided by Mrs. Barbara Kahan in connection with this simulation study.

E. B. Parker III

E. B. Parker, III

P. S. Schaenman

1031-$\frac{\text{EBP}}{\text{PSS}}$-mph

P. S. Schaenman